Proceedings of EnCon2011
4th Engineering Conference
Kuching, Sarawak, Malaysia

Applying Firefly Algorithm in Finding Optimized Path in PCB Holes Drilling Process

Asrani Lit, Amar Faiz Zainal Abidin, Muhammad Iqbal Zakaria,
Kian Sheng Lim, M. Nasir Ayob, Farhaana Yakop

Abstract – **In PCB holes drilling process, the time taken for a task completion heavily rely on the distance travels by the drill bit of the CNC machine. In order to minimize the distance traveled by the drill bit, Firefly Algorithm can be used. The proposed model applies Firefly Algorithm to search for the optimized path in PCB holes drilling process. Each agent's position represented a possible path that can be taken by the drill bit. The fitness of the agent is inversely proportional to the distance of the path where the shorter the distance, the better the fitness of an agent. Then, the agents compare their fitness with each other. Agent will try to improve its fitness by moving closer towards other agent with better fitness. The process repeated until maximum iteration achieved. Performance of the proposed model is compared with other literatures using a standard case study.**

Keywords: Computational Intelligence, Firefly Algorithm, Printed Circuit Board, Path Optimization

I. Introduction

Holes drilling process of a printed circuit board (PCB) is done using a computer numerical controlled (CNC) machine where the drill bit of the CNC machine will travel to the targeted coordinates, one at a time and drill the holes. The time spent moving the drill bit from one coordinate to another coordinate is significant, the need to optimize the PCB making time is essential, in order to increase the yield of PCB produced. Minimizing the distance travel by the drill bit can do this, as the time spent per PCB is proportional to the distance traveled. Recently developed, Firefly Algorithm (FA) is one of the algorithms available for use.

Firefly Algorithm is introduced by Xin-She Yang, which based on the flashing characteristic of the fireflies [1]. Firefly with greater flashing light intensity will attract the other fireflies to move toward it. The author translated this characteristic into a general algorithm that can be adapts into any optimization problem such as path optimization.

Path optimization in PCB holes drilling process is not a new field. Due to its importance in increasing the productivity of PCB making, many literatures had been written suggesting application of numerous optimization techniques to find the optimal path for PCB holes drilling process. In 1996, Kolahan and Liang wrote a paper proposing the use of Tabu search algorithm for variable holes sizes in holes drilling process [2]. Then, the authors extend the research by suggesting an improved algorithm in another journal [3]. Sigl and Mayer introduced the 2-Opt Heuristic Evolutionary algorithm to solve path optimization in CNC drilling problem [4]. Qudeiri *et. al*. wrote a paper on the use of Genetic Algorithm (GA) in finding the optimal path for holes cutting process using CNC machine tool [5]. While in another literatures, Kentli and Alkaya suggested the use of Record-to-Record Travel with Local Exchange Moves (RRTLEM) algorithm [6]. Applications of variations of Particle Swarm Optimization (PSO) in path optimization in PCB holes drilling process can be seen in 4 literatures. The first one is by Onwubolu and Clerc, who initiated the idea of using PSO in path optimization in holes drilling process [7]. Then, Zhu proposed new PSO algorithm called Global Convergence PSO (GCPSO) in solving local minima problem encountered by PSO [8],[9]. Next, Adam *et. al*. proposed the use of linear decrease inertia weight with PSO in tackling path optimization problem in PCB holes drilling process [10]. Recently, Othman *et. al*. applied Binary Particle Swarm Optimization (BPSO) in solving the problem [11].

This paper explores the use of Firefly Algorithm in path optimization in PCB holes drilling process. The agents (or fireflies) represent candidates' solution in a D-dimension search space. Each dimension represents path sequence need to travel by the drill bit, where the dimension value represents the hole number.

Asrani Lit is with the Faculty of Engineering, Universiti Malaysia Sarawak, 94300 Kota Samarahan, Sarawak Malaysia (e-mail: lasrani@feng.unimas.my).
Other authors are/were with the Faculty of Electrical Engineering & Faculty of Computer Science and Information System, Universiti Teknologi Malaysia, 81310 UTM, Johor, Malaysia.

II. Path Optimization Problem in PCB Holes Drilling Process

The distance of a path can be calculated using Travel Salesman Problem (TSP) formula which as stated in (1) [12].

$$c_{total} = \sum_{i=0}^{n} \sum_{j=0}^{n} \left(|x_i - x_j| + |y_i - y_j| \right) \times p_{ij} \quad (1)$$

where n is the number of holes need to be drilled. p_{ij} is the decision variable related to the movement of the drill bit from hole i to hole j. If there is a movement of the drill bit from hole i to hole j, $p_{ij} = 1$, otherwise, $p_{ij} = 0$. Coordinates of hole i is (x_i, y_j) and the coordinate of hole$j$ is (x_i, y_j). Most of PCB path problem is a symmetrical problem, thus $c_{ij} = c_{ji}$. The interest of the problem is to minimize the distance thus leads (1) to (2).

$$min(c_{total}) = min(\sum_{i=0}^{n} \sum_{j=0}^{n} (|x_i - x_j| + |y_i - y_j|) \times p_{ij}) \quad (2)$$

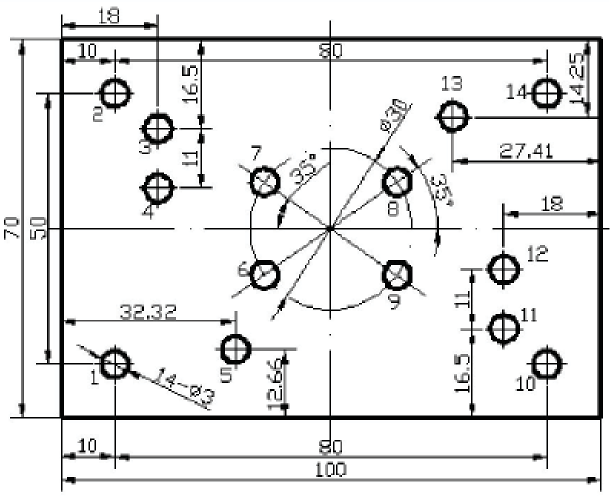

Figure 1. Case study which consists of 14 holes

The case study used in this paper, as shown in figure 1, is similar to the previous work, which has been carried out by Zhu [8]. Adam [10] and Othman [11] had used the same case study to benchmark their works.

The case study is a PCB, which consists of 14 holes. For the case study, it is assumed that all the holes are having the same size. The home position is located at the top left corner of the PCB image. A complete information of the case study as stated in table 1.

Note that Zhu, Adam and Othman had ignored the distance between home (Hole 0) to the 1st hole and distance between to the last hole to home (Hole 0) which leads (2) to (3).

$$min(c_{total}) = min(\sum_{i=1}^{n}\sum_{j=1}^{n}(|x_i - x_j| + |y_i - y_j|) \times p_{ij}) \quad (3)$$

The best solution of the case study can be seen in figure 2, where the path can be either the path having sequence 2–3–4–7–8–13–14–10–11–12–9–6–5–1 or 1–5–6–9–12–11–10–14–13–8–7–4–3–2. The distance of the best solution is 280mm.

III. Modeling Problem Using Firefly Algorithm

Firefly Algorithm was introduced by Xin-She Yang in 2007 which fundamentally based on the mating behavior of fireflies. Algorithm 1 displays the adaptation of the original Firefly Algorithm proposed by the original author [13].

In modeling path of a PCB holes drilling process, an agent's position in Firefly Algorithm represents a possible solution of the problem. The relation of agent's position with solution of the problem can be generalized as (4).

$$s_m = [Hole\ Number\ for\ 1^{st} Sequence, \dots, Hole\ Number\ for\ n^{th} Sequence]^T \quad (4)$$

Example, $\boldsymbol{s}_2 = [2,3,1]$ means that the 2nd agent proposed that the drill bit starts travel from home (Hole 0) to Hole 2, follows by Hole 3. Then, the drill bit moves to Hole 1 before returns to home. As discussion in earlier chapter, the fitness function of the problem as (5).

$$f(\boldsymbol{s}_m) = \sum_{i=1}^{n}\sum_{j=1}^{n}(|x_i - x_j| + |y_i - y_j|) \times p_{ij} \quad (5)$$

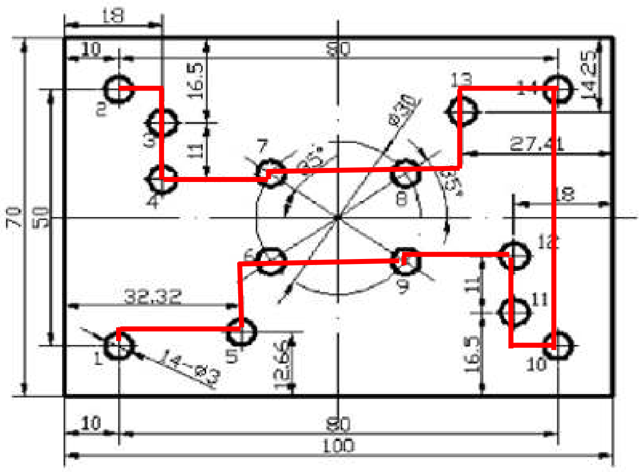

Figure 2. Optimal solution for the case study

The algorithm starts by generating initial population of agent, randomly. The agents' positions are evaluated using the fitness function in (5). Light intensity, I_m is equates to be equal to the inverse value of the agent's fitness function as shown in (6).

$$I_m = \frac{1}{f(s_m)} \quad (6)$$

From here on the algorithm will start looping until stopping criteria are fulfilled. For this study, maximum iteration, t is chosen as stopping criteria where the algorithm will stop when the iteration, z reached maximum iteration, t.

For each iteration, each agent will move toward to other agent with greater light intensity. The movement of this agent is bounded by (7).

$$\boldsymbol{s}_m = \boldsymbol{s}_m + \beta_0 e^{-\gamma r_{mu}^2}(\boldsymbol{s}_m - \boldsymbol{s}_u) + \alpha \boldsymbol{e}_m \quad (7)$$

where r is the distance between two agents in Cartesian distance. Given agent m and agent n, the Cartesian distance can be calculated using (8).

$$r_{mu} = \|\boldsymbol{s}_m - \boldsymbol{s}_u\| \quad (8)$$

β_0 is the agent's attractiveness at $r = 0$. γ is absorption coefficient. α is randomization parameter which in range [0,1]. $\boldsymbol{e}_m$ is a vector random number taken from uniform distribution.

After the agent moves towards another agent, the new position suggested might be invalid due to inexact holes number representation. Example of invalid solutions are $\boldsymbol{s}_1 = [1.23, 0.7, -7.89]$ and $\boldsymbol{s}_2 = [3, 2.91, 0.1, 89.2, -9.1]$. The proposed model suggested that the smallest value in the new position is assigned as Hole 1 and largest number is assigned to Hole n in ascending order. Using the same example, $\boldsymbol{s}_1 = [1.23, 0.7, -7.89]$ now is $\boldsymbol{s}_1 = [3, 2, 1]$, while $\boldsymbol{s}_2 = [3, 2.91, 0.1, 89.2, -9.1]$ after being corrected is $\boldsymbol{s}_2 = [4, 3, 2, 5, 1]$.

The fitness of the new agent's position is evaluated and the light intensity is updated. If the fitness obtained smaller than the global best record, the new fitness will become the new global best and the agent's position is kept as the best solution found so far.

Algorithm 1: Firefly Algorithm for Path Optimization Problem

Set fitness function, $f(\boldsymbol{s}_m)$ according to (5) where $\boldsymbol{s}_m = [s_{m1}, s_{m2}, .., s_{mn}]^T$
Generate randomly initial population of agent, $\boldsymbol{s}_m$ where $m = 1,2, .., q$
Find agent's light intensity, I_m at $\boldsymbol{x}_m$ using (6)
Define light absorption coefficient, γ
while $z < t$
  for m = 1 to q
    for u = 1 to q
      if $I_m < I_u$
        Move agent m towards u using (7)
        Perform correction if necessary
        Evaluate new solution using (5), update I_m using (6) and global best if necessary
      end if
    end for n
  end for m
end while
Post process results and visualization

TABLE I. Information of the case study

Number of holes, n		14		Length	100mm	Wide	70mm
Hole Number	Coordinate (x, y)	Hole Number	Coordinate (x, y)	Hole Number	Coordinate (x, y)	Hole Number	Coordinate (x, y)
0	(0,0)	1	(10,60)	2	(10,10)	3	(18,16.5)
4	(18,27.5)	5	(32.32,57.34)	6	(37.7,43.6)	7	(37.7,26.4)
8	(62.3,26.4)	9	(62.3,43.6)	10	(90,60)	11	(82,53.5)
12	(82,42.5)	13	(72.59,14.25)	14	(90,10)		

TABLE II. Comparison of the PSO parameters used by several literatures with this study

	Zhu's [8]	Adam [10]	Othman [11]	This study
Common Parameters				
Number of agents, q	100	50	50	50
Number of iterations, t	10000	5000	2500	10000
Number of computations	50	50	50	50
PSO Parameters				
Inertia weight, ω	0.0, 0.5, 1.0	0.9 → 0.4		Not applicable
Cognitive component, c_1	Not available	1.42		Not applicable
Social component, c_2	Not available	1.42		Not applicable
r_1 and r_2	Random number [0,1]			Not applicable
FA Parameters				
Attractiveness, β_0	Not applicable			1
Randomization parameter, α	Not applicable			1
Absorption coefficient, γ	Not applicable			1

TABLE III. Comparison PSO resulted obtained by Zhu and this study

	Zhu's [8]			Adam [10]	Othman [11]	This study
	GCPSO			PSO	BPSO	FA
Inertia weight, ω	0	0.5	1.0	0.9 → 0.4		Not Applicable
The least iteration number while global convergence	70	601	93	118	71	22
The average iteration number while global convergence	1784	3549	2104	1415	783	1652.4
Length of optimal solution	280	280	280	280	280	280
Average fitness after computing 50 computations	305.7	307.3	289.6	292.3	296.0	288.2

IV. IMPLEMENTATION AND EXPERIMENTAL RESULT

The program is written in Matlab and the simulation is performed 50 times on a desktop that equipped with 1.86GHz Intel Pentium Core 2 processor with 2GB RAM. Table 2 listed out the parameters values used by several literatures and this study for solving the case study. Table 3 tabulated the result obtained by other literatures versus to this study. Each computation takes an average duration of 1.5 hours.

From table 3, it can be seen that the proposed model has a lower average fitness after 50 computations compared to PSO, GCPSO and BPSO. Lower average fitness translates to a better choice of path selection by the CNC machine in completing its task. Much more important point is that the proposed model managed to find optimal solution like PSO, GCPSO and BPSO.

The average iteration number while global convergence for the proposed model is larger than PSO and BPSO. A really small value of average iteration with poor average fitness value might suggest that the model converge prematurely. This speculation might applicable to Adam's PSO and Othman's BPSO.

V. CONCLUSION

This paper has proposed and studied the use of Firefly Algorithm in searching optimized path for PCB holes drilling process. The result shows that the proposed model performance is better than previous literatures. Further studies on the sensitivity of the parameters in Firefly Algorithm should be conducted. This will improve the performance of the proposed approach.

REFERENCES

[1] X. S. Yang, "Firefly Algorithm, Stochastic Test Functions and Design Optimisation," *International Journal Bio-Inspired Computation*, vol. 2, no. 2, pp.78–84, 2010.

[2] F. Kolahan, M. Liang, "Tabu search approach to optimization of drilling operations," *Computers and Industrial Engineering*, vol. 1-2, no. 31, pp. 371-374, 1996.

[3] F. Kolahan, M. Liang, "Optimization of hole-making operations: a tabu search approach," *International Journal of Machine Tools and Manufacture*, vol. 2, no. 40, pp. 1735-1753, 2000.

[4] S. Sigl, H. A. Mayer, "Hybrid Evolutionary Approaches to CNC Drill Route Optimization," *Proceedings of Computational Intelligence for Modeling, Control and Automation*, pp. 905-910, 2005.

[5] J. A. Qudeiri, H. Yamamoto, R. Ramli, "Optimization of operation sequence in CNC machine tools using genetic algorithm," *Journal of Advanced Mechanical Design, Systems, and Manufacturing*, vol. 1, no. 2, pp. 272-282, 2007.

[6] A. Kentli, A. F. Alkaya, "Deterministic Approach to Path Optimization Problem," *Ozean Journal of Applied Sciences,* vol.2, no. 2, pp. 149-157, 2009.

[7] G. C. Onwubolu, M. Clerc, "Optimal path for automated drilling operations by a new Heuristic approach using particle swarm optimization," *International Journal of Production Research*, vol. 3, no. 42, pp. 473-491, 2004.

[8] G. Y. Zhu, "Drilling Path Optimization Based on Swarm Intelligent Algorithm," *Proceedings of IEE International Conference on Robotics and Biomimetics*, pp. 193-196, 2006.

[9] G. Y. Zhu, W. B. Zhang,"Drilling path optimization by the particle swarm optimization algorithm with global convergence characteristics," *International Journal of Production Research*, vol. 46, no. 8, pp. 2299-2311.

[10] Jing Zhang, "Natural Computation for the Traveling Salesman Problem", *Proceeding Second International Conference on Intelligent Technology and Automation, IEEE Computer Society*, Vol.1, pp.366 - 369, 2009.

[11] A. Adam, A. F. Z. Abidin, Z. Ibrahim, A. R. Husain, Z. M. Yusof, I. Ibrahim, "A Particle Swarm Optimization Approach to Robotic Drill Route Optimization," *Fourth Asia International Conference on Mathematical/Analytical Modelling and Computer Simulation*, pp. 60-64, 2010.

[12] M. H. Othman, A. F. Z. Abidin, A. Adam, Z. M. Yusof, Z. Ibrahim, S. M. Mustaza, L. Y. Yang, "A Binary Particle Swarm Optimization Approach for Routing in PCB Holes Drilling Process," *1st International Conference on Robotic Automation System*, pp.201-206, 2011.

[13] X. S. Yang, *Engineering Optimization*. Hoboken, New Jersey: Wiley, 2010, pp. 221-230.